

Partial translation of JP9-6801A

[0027] In the general database management system, metadata of names of stored tables and columns are managed by a table called catalogue (or data dictionary), making the data accessible by using a general SELECT sentence. Names and structures of the catalogue tables are, however, different by database vendors. For this reason, as the above-described functional interface, there have been provided interfaces independent of database vendors, while designating functions specialized for access to the catalogues.

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-6801

(13) 公開日 平成9年(1997)1月10日

(5) int.CI ^b	識別記号	庫内整理番号	F I	技術表示箇所
G 0 6 F	17/30	9289-5L	G 0 6 F	15/40
	12/66	5 1 2	7623-5B	12/00
			9289-5L	15/403
			9289-5L	15/413

審査請求 来請求 請求項の数? 81 (全 13 頁)

(21)出願番号	特願平7-153410	(71)出願人	000005108 株式会社日立製作所 東京都千代田区神田穀町四丁目6番地
(22)出願日	平成7年(1995)6月20日	(72)発明者	翁地 雄 神奈川県川崎市麻生区王禅寺1090番地 株式会社日立製作所システム開発研究所内 高橋 由美子
		(72)発明者	神奈川県川崎市麻生区王禅寺1090番地 株式会社日立製作所システム開発研究所内 玄蕃 錠
		(72)発明者	神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア開発本部内
		(74)代理人	井理士 富田 和子

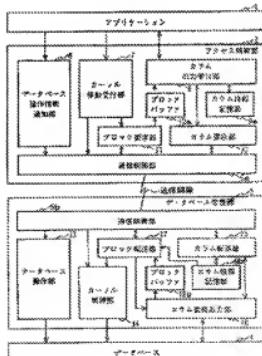
(54) 【発明の名称】 データベースアクセスシステム

《57》【秦韵】

【目的】高速にデータベースをアクセス可能であり、さらに、データの柄落ち、丸め等が無い、データベースのアクセスシステムを操作すること。

【構成】第1カラム情報記憶手段は、属性を記憶する。カラム選択手段は、属性をデータベース管理手段に通知し、1つのカラムデータの出力要求を発行し、1つのカラムデータを受け取る。ブロック要求手段は、1以上のカラムデータの一括しての出力要求を発行し、カラムデータを受け取る。カラム出力条件手段は、カラム要求手段が受け取ったカラムデータまたはブロック要求手段が受け取ったカラムデータの内の1カラムデータを出力する。カラム転送手段は、出力要求命令に応じ、属性を参照し、カラムデータを受け取る。アクセス制御手段へ転送する。ブロック転送手段は、出力要求命令に応じ、1以上のカラムデータを取得し、アクセス制御手段へ転送する。

卷二



【特許請求の範囲】

【請求項1】データベースの記憶データを取得するシステムであつて、

1種類以上のカラムデータを有するレコードを複数備えているデータベースと、カラムデータを取得するデータベース管理手段と、与えられた出力要求に対して、前記データベース管理手段のカラムデータ取得動作を制御するアクセス制御手段とを備え。

前記アクセス制御手段は、

前記与えられた出力要求の対象となるカラムデータの出力属性を規定する情報である属性を記憶する第1カラム情報記憶手段と、前記属性を前記データベース管理手段に通知しつつ、1つのカラムデータの出力要求の命令を発行する機能、および、1つのカラムデータを受け取る機能を有するカラム要求手段と、1つ以上のカラムデータの、一括しての出力要求の命令を発行する機能。および、1つ以上のカラムデータを一括して受け取る機能を有するブロック要求手段と、前記カラム要求手段が受け取った1つ以上のカラムデータの内の1つのカラムデータ、を出力するカラム出力受付手段とを有し、

また、前記データベース管理手段は、

前記カラム要求手段により通知された前記属性を記憶する第2カラム情報記憶手段と、前記カラム要求手段からの出力要求命令に応じ、前記データベースからカラムデータを取得し、前記アクセス制御手段へ転送するカラム転送手段と、前記ブロック要求手段からの出力要求命令に応じ、前記属性を参照して前記データベースから1つ以上のカラムデータを取得して、一括して前記アクセス制御手段へ転送するブロック転送手段とを有する、データベースアクセスシステム。

【請求項2】請求項1において、前記カラム出力受付手段は、

出力要求されたカラムデータが属するカラムに対しての最初の出力要求に応じた場合、または、出力要求に応じ既に出力したカラムデータが属するレコードを前レコードとした時、前レコードの内で出力したカラムデータと、同一カラムに属するカラムデータに対する出力要求であつて、出力時のデータ形式が前レコードとは異なる場合。

前記カラム要求手段を起動して、出力要求の命令を発行する手段を、備えたことを特徴とするデータベースアクセスシステム。

【請求項3】請求項1において、前記カラム出力受付手段は、

出力要求に応じ既に出力したカラムデータが属するレコードを前レコードとした時、前レコードの内で出力したカラムデータと、同一カラムに属するカラムデータに対する出力要求であり、かつ、出力時のデータ形式に変更がない場合。

前記ブロック要求手段を起動して、出力要求の命令を発行する手段を、備えたことを特徴とするデータベースアクセスシステム。

【請求項4】請求項1において、前記ブロック転送手段は、

前記第2カラム情報記憶手段に記憶された属性を参照して、カラムデータを所定のデータ形式に変換する処理を、1以上のカラムデータに対して行なう手段と、前記処理が行なわれたカラムデータを一括して前記アクセス制御手段へ転送する手段とを、備えたことを特徴とするデータベースアクセスシステム。

【請求項5】請求項1において、前記ブロック転送手段が一括して転送する1つ以上のカラムデータは、

出力要求に応じて既にに出力したカラムデータが属するレコードを前レコードとした時、前レコードの内で出力した、1以上のカラムデータと、同一カラムに属する1以上他のカラムデータからなり、かつ、各カラムデータは、列に対する、前レコードの内で出力したカラムデータと、同一のデータ形式を有するデータである、ことを特徴とするデータベースアクセスシステム。

【請求項6】請求項1において、前記属性は、出力カラムの種別情報を、出力時のデータ形式識別情報を有する情報である、ことを特徴とするデータベースアクセスシステム。

【請求項7】請求項1において、与えられた命令によつて、データベースの記憶データに対して、所定の処理を行なうデータベース操作手段と、与えられた命令を変換する命令変換部とを備え。

前記データベース操作手段は、前記命令変換部によって変換された命令を、前記与えられた命令として受け付ける機能を有する、データベースアクセスシステム。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、データベースの記憶内容を取得するためのシステムに係り、特に、オーバーヘッドの低減を図り、リレーションナルデータベースのアクセスに適応したシステムを提供するための技術に関するもの。

【0002】

【従来の技術】近年、様々な種類のデータを共有し、情報の有効活用を図るためにデータベース管理システムが、提案されている。現在提案されている各種のデータベース管理システムの中でも、データの譲達付けを考慮したテーブルを利用してデータを取り扱う、リレーションナルデータベース管理システムが急速に普及している。

【0003】リレーションナルデータベースでのデータ記憶の基本的な考え方は、テーブルを利用することによつて行なう。

【0004】テーブルは、カラム(列)とレコード(行)からなる二次元の表であり、表を構成する各要素

は、カラムデータと称されている。また、通常、カラム(列)は複数存在するので、1行、即ち、1レコードは、複数種類のカラムデータによって構成されている。

【0005】このようなリレーションナルデータベースの管理システムは、今日のビジネス社会に不可欠なシステムとなっており、ISO(国際標準化機構)は、リレーションナルデータベース用言語である「SQL(Structured Query Language)」を規定している。

【0006】図2に、SQL構文のシンタクスを示す。

【0007】SELECT文29は、データベース上のテーブルを検索するSQL構文であり、(SELECT、IFROM、WHERE等)のキーワードを有して構成される。

【0008】具体的には、キーワード(=SELECT)の後に、カラムデータの出力要求対象となる、1以上のカラム名が記述される。検索対象となるテーブルの名称を表す「テーブル名」は、カーワード(IFROM)の後に記述される。さらに、検索条件は、キーワード(WHERE)の後に、記述される。

【0009】このような、SELECT文29をリレーションナルデータベース管理システムへ発行することにより、導出表が得られることになる。ここで「導出表」とは、データベースに格納されたテーブル(実表)に対して、実義を部分的に切り出した論理的なテーブルであり、切り出されるデータは、SELECT文29中のカラム名群、および、検索条件により決まることになる。

【0010】次に、FETCH文30は、導出表中の1レコードを取り出し、カーソルを、次のレコードに移動させるSQL構文である。ここで、「カーソル」とは、導出表中の1レコードを論理的に指し示す、一種のポインターとして機能するものであり、該ポインターは、レコードを単位として移動する。導出表から取り出された1レコードは、各カラム毎に、FETCH文30で指定されたホスト変数に格納される。アプリケーションは、FETCH文30を複数回発行することにより、導出表を構成する全レコードを得ることになる。

【0011】図5に、データベースへのアクセスに関する従来の情報シーケンスを示す。

【0012】図5において、1は、ユーザーのアプリケーション、2は、データベースへのアクセスを制御するアクセス制御部であり、3は、データベースを管理するデータベース管理部である。なお、近年のクライアント・サーバシステムの発達に鑑み、アプリケーション1およびアクセス制御部2を備えるクライアントと、データベース管理部3を接続するサーバとをネットワークで接続し、クライアント・サーバシステムによって、データベースのアクセス権能を有するシステムが提案されつつある。

【0013】図5に示すように、まず、アクセス制御部

2は、アプリケーション1から発行されたSELECT文29を受け付けて、データベース管理部3に対し、導出表の作成を依頼するアクセス命令を発行する。次に、アクセス制御部2は、アプリケーション1から発行されたFE TCH文30を受け付けて、データベース管理部3に対して、複数のカラムデータの転送を依頼するブロック出力要求を発行する。ブロック出力要求を受け付けたデータベース管理部3は、現在のカーソルが指示する1レコードを構成する、全カラムデータを取り出し、ブロックデータとして一括して、アクセス制御部2へ返送する。

【0014】そして、1レコード分のブロックデータを受信したアクセス制御部2は、FE TCH文30で、受信したブロックデータ中の各カラムデータを、対応する、予め指定された各ホスト変数に格納する。このような動作を繰り返して、アプリケーション1は、所望のデータを得ることになる。

【0015】なお、データの転送時間を削減するため、特開平3-189741号公報に記載されているように、全レコードを予め一括して転送する方法も提案されている。

【0016】ところで、このような従来のシステムのユーザーは、上記SQL構文を、データベース特有の言語(例えば、埋め込みSQL)を使用し、データベース用のアプリケーションを開発してきた。

【0017】しかしながら、今日では、通常のプログラミング言語との親和性を向上させるため、呼び出し関数(以下、単に、「関数」と称することが多い)を用いた、データベースアクセスインターフェース(CLI: Call Level Interface)が、規定されている。ユーザーは、このインターフェースを用いることにより、使い慣れたプログラミング言語を使用して、データベース用のアプリケーションを開発することができる。

【0018】図3に、関数を用いた、データベースアクセス用のインターフェースの一例を示す。

【0019】SQLExecute関数31は、データベース上のテーブルを検索するSELECT文等のSQL構文を指定するための関数であり、パラメータとしてSQL文の文字列を採用する。

【0020】SQLFetch関数32は、カーソルを移動させるための関数である。

【0021】SQLGetCol関数33は、カーソルが、現在、指し示すレコード中の1つのカラムデータを抽出するための関数であり、パラメータとしては、カラム番号25、データ型26、バッファアドレス27、データ長アドレス28がある。

【0022】カラム番号25は、導出表中の個々のカラムを識別するための連続番号であり、通常、左端のカラム(導出表中で最初に出現するカラム)に「1」が割り当てられる。

【0023】データ型26は、出力要求に対してデータ

を出力する時の、データ型（本明細書において、「データ形式」と表現する場合もある）を表す識別子である。一般に、データベースが管理可能なデータ型は、プログラミング言語で扱うデータ型より広範（例えば、浮動小数点データ型における精度等）である。アプリケーションは、データ型を指定することにより、データベースに格納されているデータを、所望のデータ型に変換して出力させることができる。

【0024】バッファアドレス27は、出力データを格納するための領域のアドレスであり、また、データ長アドレス28は、出力したデータの長さを格納する領域のアドレスである。

【0025】上記関数を用いたデータベースアクセス用のインターフェースを使用するアプリケーションは、SQL文を発行するSQLExec関数31、および、SQLFetch関数32を発行した後、出力要求の対象となるカラム毎に、SQLGetCol関数33の発行を繰り返し、1レコードのデータを取得する。

【0026】現レコード中の、出力要求の対象となるカラムのカラムデータを全て取得した後、SQLFetch関数32を発行することにより、カーソルを次のレコードに移動し、再び、SQLGetCol関数33の発行を繰り返すことになる。

【0027】また、一般的なデータベース管理システムでは、格納されたテーブルやカラムの名前等のメタデータが、カタログ（または、データダイクショナリ）と称されるテーブルで管理されており、通常のSELECT文を用いて、データのアクセスが可能である。しかしながら、カタログテーブルの名称やテーブル構造等は、データベースペンド毎に異なる。そこで、上記関数インターフェースとして、カタログアクセスのため特別な関数を規定し、データベースペンドに依存しないインターフェースを提供している。

【0028】従来のアクセス制御部2、または、データベース管理部3は、前記関数の処理要求を受けて、独自のカタログテーブルを、予め定義された構造で「射影（テーブルから特定のデータを取り出す）」して導出表を作成する。この場合、導出表のカラム構成が静的であるため、アプリケーション1は、カタログアクセス関数を発行した後、所望のカラムに設定してSQLGetCol関数33を発行することで、必要なカラムデータだけ入手していた。

【0029】

【発明が解決しようとする課題】ところで、以上のような、関数を用いたデータベースアクセス用のインターフェースを、上記従来の方法で処理すると、以下のような問題が発生する。

【0030】従来のアクセス制御部2は、アプリケーション1からSQLFetch関数32を受け付けた時点で、1レコードのカラムデータを一括して、データベース管理シ

ステム3から取り出している。ところが、この時点では、アプリケーション1が各々のカラムを、どのようなデータ型で出力させたいか判断できないため、ブロック出力要求を受けたデータベース管理部3は、予め定められたデータ型に変換処理してカラムデータを取り出し、一括してアクセス制御部2へ返送する。アクセス制御部2は、SQLGetCol関数33を受け付けた時点で、受信したカラムデータをアプリケーション1から指定されたデータ型に再変換し、アプリケーション1へ返す。

【0031】しかしながら、一度、あるデータ型に変換した後、他のデータ型へ再変換すると、数据データ型では、桁落ち、精度の丸め等が発生する場合があり、システムの信頼性を低下させることになる。

【0032】一方、全てのデータ型を、文字列データ型に変換して取り出し、SQLGetCol関数33を受けた時点でのアプリケーション1が所望するデータ型へ再変換するように変換処理すれば、桁落ち、精度の丸め等の発生を、ある程度抑制することもできる。しかし、数値型のデータを文字列に変換するとデータ量が増えるため、データ転送のオーバヘッドが増加する。さらに、数値と文字列との間の変換処理に要するオーバヘッドもかかり、データベースシステムのアクセス時間等の性能に影響を与える。

【0033】また、上記従来技術によると、データベース管理部3は、SELECT文を含んだアクセス命令を受けて、導出表を作成する。アクセス制御部2は、アプリケーション1からSQLGetCol関数33を受け付ける前に、データベース管理部3が作成した導出表から、1レコードを構成する、全カラムデータを取り出す。通常、データベースを検索する場合には、SELECT文を使用して必要なカラム名を特定できることから、アプリケーション1は、導出表の全カラムに対して出力を要求するケースが多く、窓もって導出表の全カラムデータを取り出しても、性能に悪影響を及ぼすことは少ない。

【0034】ところが、上記関数を用いたインターフェースを使用して、カタログをアクセスする場合、従来の方法によると、以下の問題も発生する。

【0035】すなわち、アクセス制御部2は、アプリケーション1からSQLGetCol関数33を受け付ける前に、データベース管理部3に対して、ブロック出力要求を発行する。この時点で、アクセス制御部2は、アプリケーション1がカタログ中のどのカラムの出力を要求しているか判断できないため、1レコードの全カラムデータをデータベース管理部3から取り出し、以降、アプリケーション1から受けたSQLGetCol関数33に応じて、個々のカラムデータを、アプリケーション1に返す処理を行なう。しかしながら、カタログアクセス要求の場合には、アプリケーション1が導出表の全カラムに対して出力を要求するとは限らない。このとき、上記従来の方法によると、アプリケーション1が望まない冗長なカラム

データも、アクセス制御部2に転送されることになり、データベースシステムのアクセス性能が劣化してしまう。

【0036】上記のような従来の方法における各課題は、図4に示すように、リモートアクセス制御部2が、アプリケーション1からの処理要求に対して、各自に対応する処理要求をデータベース管理手段3へ発行することで解決することもできるが、この方法では、例えば、「mカラム×nレコード」の導出表を発行する際、「m×n回」のカラム出力要求が発行されることになり、処理オーバーヘッドが増大しデータベースのアクセス性能を大幅に劣化させることになっていた。

【0037】そこで、本発明の目的は、処理オーバーヘッドの低減や、裕度ら、精度のため等の発生を防止し、高速かつ高精度性を有する、データベースアクセスシステムを提供することにある。

【0038】

【課題を解決するための手段】上記課題を解決し、本発明の目的を達成するため、以下の手段が考えられる。

【0039】すなわち、データベースの記録データを取得するシステムであって、1種類以上のカラムデータを有するレコードを複数取得しているデータベースと、カラムデータを取得するデータベース管理手段と、与えられた出力要求に対して、前記データベース管理手段のカラムデータ取得動作を制御するアクセス制御手段とを備える。

【0040】そして、前記アクセス制御手段は、前記与えられた出力要求の対象となるカラムデータの出力機能を規定する情報を有する属性を記憶する第1カラム情報記憶手段と、前記属性を前記データベース管理手段に通知しつつ、1つのカラムデータの出力要求の命令を発行する機能、および、1つのカラムデータを受け取る機能を有するカラム要求手段と、1つ以上のカラムデータの一括しての出力要求の命令を発行する機能、および、1つ以上のカラムデータを一括して受け取る機能を有するブロック要求手段と、前記カラム要求手段が受け取ったカラムデータ、または、前記ブロック要求手段が受け取った1つ以上のカラムデータの内の1つのカラムデータ、を出力するカラム出力受付手段とを有した構成にする。

【0041】また、前記データベース管理手段は、前記カラム要求手段により通知された前記属性を記憶する第2カラム情報記憶手段と、前記カラム要求手段からの出力要求命令に応じ、前記データベースからカラムデータを取得し、前記アクセス制御手段へ転送するカラム転送手段と、前記ブロック要求手段からの出力要求命令に応じ、前記属性を参照して前記データベースから1つ以上のカラムデータを取得して、一括して前記アクセス制御手段へ転送するブロック転送手段とを有した構成にする。

【0042】

【作用】データベースには、1種類以上のカラムデータを有するレコードを複数持てておき、データベース管理手段は、データベースからカラムデータを取得する。アクセス制御手段は、与えられた出力要求に対して、前記データベース管理手段のカラムデータ取得動作を制御する。

【0043】さらに、具体的には、アクセス制御手段の各種構成要素は、以下のように動作する。

【0044】まず、第1カラム情報記憶手段は、前記与えられた出力要求の対象となるカラムデータの出力態様を規定する情報を記憶する。

【0045】カラム要求手段は、前記属性を前記データベース管理手段に通知しつつ、1つのカラムデータの出力要求の命令を発行し、また、1つのカラムデータを受取る。

【0046】これに対し、ブロック要求手段は、1つ以上のカラムデータの、一括しての出力要求の命令を発行し、また、1つ以上のカラムデータを一括して受け取る。さらに、カラム出力受付手段は、前記カラム要求手段が受け取ったカラムデータ、または、前記ブロック要求手段が受け取った1つ以上のカラムデータの内の1つのカラムデータ、を出力する。

【0047】また、データベース管理手段の各種構成要素の動作は以下のようになる。

【0048】まず、第2カラム情報記憶手段は、前記カラム要求手段により通知された前記属性を記憶する。カラム転送手段は、前記カラム要求手段からの出力要求命令に応じ、前記データベースからカラムデータを取得し、前記アクセス制御手段へ転送する。一方、ブロック転送手段は、前記ブロック要求手段からの出力要求命令に応じ、前記属性を参照して前記データベースから1つ以上のカラムデータを取得して、一括して前記アクセス制御部へ転送する。

【0049】本発明によれば、アプリケーション等から要求されたカラムデータのみを、要求されたデータ形式で転送するため、高速かつ高精度な、データベースのアクセス動作を実現可能とする。

【0050】

【実施例】以下、本発明の実施例について、図面を参照しつつ説明する。

【0051】なお、以下の図中、同一の構成要素には、同一の符号を付する。

【0052】図1は、本実施例にかかるデータベースアクセスシステムの構成図である。

【0053】1は、ユーザーのアプリケーション、2は、アクセス制御部、3は、データベース管理部、4は、データベースであり、アクセス制御部2とデータベース管理部3は、ネットワークで接続されている。

【0054】なお、クライアント・サーバシステムを構

策するとき、クライアント側が、アクセス制御部2を備えた構成とし、また、サーバ側がデータベース4とデータベース管理部3を備えるようにする構成が考えられる。このとき、クライアントを複数台設けてシステムを構築しても良いことは、言うまでもない。

【0055】データベース4は、例えば、1種類以上のカラムデータを有するレコードを複数個用意し、テーブル状(カラムを列、レコードを行とする)に格納する、いわゆるリレーションナルデータベースを採用して構築すれば良い。

【0056】アクセス制御部2は、ユーティリティが使用するアプリケーション1から、データベースを定義・操作するためのSQL文を受け付けるデータベース操作情報通知部6と、SQLFetch函数3を受け付けるカーソル移動受付部7と、SQLGetCol函数3を受け付けるカラム出力受付部8と、アプリケーション1が発行したSQLGetCol函数3で指定された属性(例えば、カラム番号等、データ型等の情報を含む)を記憶するカラム情報記憶部9aと、データベース管理部3に対して、1カラムのデータ出力要求を発行するカラム要求部10と、データベース管理部3に対して、複数のカラムデータの出力要求を発行するブロック要求部11と、ブロック要求部11が受け取ったデータブロックを一時的に記憶するためのブロックバッファ12と、データベース管理部3との間で、通信回線を介して所定のデータを通信するための通信制御部5aと、を有して構成されている。

【0057】なお、データベース操作情報通知部6、カーソル移動受付部7、カラム要求部10、および、ブロック要求部11は、いずれも通信制御部5を介して、データベース管理部3との間でデータを送受信する機能を有する。また、カラム情報記憶部9aに記憶された属性は、データベース管理部3に通知される構成になっている。

【0058】次に、データベース管理部3の構成について示す。

【0059】データベース管理部3は、データベース4上のデータを検索、更新する機能を有するデータベース操作部13と、先に説明した「導出表」中の1レコードを指示する「カーソル」の移動を制御するカーソル制御部14と、データベース4上のカラムデータを取り出し、指定されたデータ型に変換して出力するカラム変換出力部15と、アクセス制御部2から通知されたカラム出力要求に対応する属性を記憶するカラム情報記憶部9bと、カラム変換出力部16が出した1カラムデータをアクセス制御部2へ転送するカラム転送部15と、カラム変換出力部16が出した複数のカラムデータを一時的に記憶するブロックバッファ12と、ブロックバッファ12に記憶した複数のカラムデータを、一括してアクセス制御部2へ転送するブロック転送部17と、アクセス制御部2との間で、通信回線を介して所定のデ

ータを通信するための通信制御部5bと、を有して構成される。

【0060】なお、データベース操作部13、カーソル制御部14、カラム転送部16、および、ブロック転送部17は、いずれも通信制御部5bを介して、アクセス制御部2との間でデータを送受信する機能を有する。

【0061】図6に、アクセス制御部、および、データベース管理部3を構成する、カラム情報記憶部9(9a、9b)のデータ構造を示す。

【0062】カラム情報記憶部9は、アプリケーション1が、出力することを要求した複数のカラムに関する属性を、記憶可能な手段であり、1カラムに関する属性領域は、出力を要求したカラムの番号を記憶するカラム番号記憶領域18と、出力時のデータ型を記憶するデータ型記憶領域19とから構成される。

【0063】ここで、属性とはカラムデータの出力態様を定める情報であり、例えば、カラム番号や、出力データのデータ型等が挙げられる。即ち、属性は、いかなる態様でデータを出力するのかを定める情報である。

【0064】図7は、アクセス制御部、および、データベース管理部3を構成する、ブロックバッファ12(12a、12b)内の、情報の記憶状態の説明図である。

【0065】ブロックバッファ12は、複数のカラムデータに関する情報を記憶する手段であり、1つのカラムデータに対して、データの実体22に加え、導出表中のカラムの番号20、および、データの長さ22、の情報を備えている。

さて、ここで、本実施例にかかるデータベースアクセスシステムの概要を説明しておくことにする。

【0066】本実施例にかかるデータベースアクセスシステムは、アプリケーション1が、出力を要求するカラム、および、出力時のデータ型を、前レコードに対する出力要求に関する情報を参照して予測し、この予測結果に応じた出力を行なう。すなわち、前レコードで出力したカラムデータと同一種類のカラムデータを、同一のデータ型に変換した複数のカラムデータを、一括して転送する処理を行なう。これにより、高速かつ高信頼性を有するデータベースアクセスシステムを実現可能になる。なお、カーソルは、レコード単位で移動してレコードを指示していくが、現在のレコードをカーソルが指示する前に、カーソルが指示していたレコードを「前レコード」と表現している。

【0067】本実施例にかかるアクセス制御部2は、アプリケーション1から受けたSQLGetCol函数3が、導出表の、あるカラムに對する最初の出力要求である場合、または、前レコードで出力したカラムと同一のカラムに對する要求であるが、出力時のデータ型が、前レコードとは異なる場合に限り、統合して、カラムデータの出力要求をデータベース管理部3へ発行し、転送された1つのカラムデータを、アプリケーション1へ返す処理を

行なう。

【0068】上記のいずれでもない場合、即ち、前レコードで出力したカラムと同一のカラムに対する要求であり、かつ、出力時のデータ型も前レコードと同一である場合には、アクセス制御部2は、ブロック出力要求によって、前もって受け取った1レコード分のカラムデータから、目的のカラムデータを取り出してアプリケーション1へ返す処理を行なう。

【0069】本実施例にかかるデータベース管理部3は、アクセス制御部2から受けたカラム出力要求を解析し(具体的には、関数のパラメータを調べ)、カラム番号と、出力時のデータ型と、カラム情報記憶部9bに記憶する。データベース管理部3は、アクセス制御部2から、ブロック出力要求を受けて、カラム情報記憶部9bに記憶した、カラム番号とデータ型を参照し、データベース4上のカラムデータを取り出して、所定の変換を行ない、複数のカラムデータを連結して、アクセス制御部2へ返す処理を行なう。

【0070】すなはち、アクセス制御部2が発行するカラム出力要求は、データベース4から1つのカラムデータを取りだし、変換出力させるとともに、データベース管理部3へ、アプリケーション1が要求した、カラムの番号と出力時のデータ型を通知する機能を有する。

【0071】データベース管理部3は、これから出力するレコードを構成する各カラムデータが属するカラムが、前レコードで出力したカラムデータが属するカラムと同一で、かつ、各々のカラムデータが、前レコードと同一のデータ型で出力されるを仮定し、アプリケーション1からのSQLGetCol関数3を受けて待機する前に、前レコードと同一の複数カラムに属するカラムデータを、各々前レコードと同一のデータ型に変換し、一括して送信する。

【0072】図8に、本実施例におけるシステムが動作した場合の情報シーケンスを示す。

【0073】アプリケーション1は、各レコードに対して、同一のカラムに属するカラムデータを、同一のデータ型で出力することを要求する。

【0074】アクセス制御部2は、アプリケーション1から、最初のレコードに関する複数のSQLGetCol関数3を受けて、データベース管理部3へ、各々の関数に対するカラム出力要求を発行して、1カラム毎にデータを受け取って、アプリケーション1に渡す。次に、続くレコードへのSQLFetch関数3を受けて、データベース管理部3に對し、ブロック出力要求を発行し、そのレコード中の全カラムデータを一括して受け取る。

【0075】その後、アクセス制御部2は、アプリケーション1から、SQLGetCol関数3を受けて、データベース管理部3から受け取った全カラムデータの中から、要求されたカラムデータを取り出し、アプリケーション1に渡す。

【0076】ところで、図8は、アプリケーション1が、各レコードに対して、同一のカラムに属するカラムデータを、同一のデータ型で出力することを要求する場合における、情報シーケンスである。ところが、閑数を用いたインターフェースである、SQLGetCol関数3等を使用するアプリケーションは、各レコード毎に異なる出力形式、例えば、出力カラムの追加や出力時のデータ型の変更等を要求することも可能である。

【0077】本発明にかかるデータベースアクセスシステムは、このような出力要求にも容易に対応可能である点も特徴となっている。

【0078】図13は、最初のレコードで、2カラム(2種類のカラムデータ)を出力した後、以降のレコードでは、出力するカラムデータの種類を1種類追加するアプリケーションを想定した情報シーケンスである。

【0079】アクセス制御部2は、アプリケーション1から、最初のレコードに関する2つのSQLGetCol関数3を受けて、データベース管理部3へ、2つのカラム出力要求を発行し、データを受け取り、アプリケーション1に渡す。次に、続くレコードへのSQLFetch関数3を受けて、データベース管理部3に対しデータ出力要求を発行し、データを受け取り、アプリケーション1に渡す。次に、続く、レコードへのSQLFetch関数3を受けて、データベース管理部3に對し、ブロック出力要求を発行し、そのレコード中の3カラムデータを一括して受け取る。その後、アクセス制御部2は、アプリケーション1から、SQLGetCol関数3を受けて、データベース管理部から受け取った3カラムデータの中から、要求されたカラムのデータを取り出し、アプリケーション1に渡す。

【0080】さて、以下に、本実施例にかかるデータベースアクセスシステムの動作を詳細に説明する。

【0081】図9は、アクセス制御部2におけるカーソル移動受付部7の動作を示すフローチャートである。

【0082】カーソル移動受付部7は、現在、カーソルが指している位置が、導出表の先頭を指示しているか否かの情報を保持するカーソル位置記憶領域(図示せず)を備える。

【0083】例えば、カーソルが導出表の先頭を指している場合は「零」、その他は「正」の値を保持するものとする。カーソル位置記憶領域およびカラム記憶部9は、SQLExec関数3を受けて付けた時点で「零」に初期

化されるように構成されている。

【0085】まず、カーソル移動受付部7は、アプリケーション1から、SQLFetch関数32を受け付けると(S901)、カーソル位置記憶領域の記憶内容を参照し、現在のカーソル位置が最初のレコード(第1レコード)を指示しているか否かを調べる(S902)。

【0086】カーソル位置が、最初のレコードを指示している場合には、データベース管理部3へカーソル移動要求を発行し、カーソル位置記憶領域に「正」の値を登録する(S903)。

【0087】S903において、カーソル位置が最初のレコードではないと判断された場合には、ブロック要求部11へ、ブロック転送を指示する。ブロック要求部11は、データベース管理部3へブロック出力要求を発行し、一括転送された結果のカラムデータをブロックバッファ12aへ格納する(S904)。

【0088】以上が、カーソル移動受付部7が行なう動作の内容である。

【0089】図10は、アクセス制御部2におけるカラム出力受付部8が行なう動作内容を示すフローチャートである。

【0090】カラム出力受付部8は、アプリケーション1から、SQLGetCol関数3を受け付け、要求内容を示すパラメータであるカラム番号25とデータ型26とを抽出する(S1001)。次に、カラム情報記憶部9aを参照し(S1002)、受け付けたSQLGetCol関数3のパラメータが、既に登録されているか否かを判断する(S1003)。

【0091】まず、カラム出力受付部8は、抽出したカラム番号25と同一の値が登録されたカラム番号記憶領域18が存在するか否かを確認し、存在しない場合には、新たなカラム番号記憶領域18とデータ型記憶領域19に、抽出したパラメータを登録する(S1004)。

【0092】存在を確認した場合、さらに、そのカラム番号記憶領域18に対応するデータ型記憶領域19の値が、要求内容から抽出したデータ型26と同一であるか否かを確認する。同一でない場合には、データ型19の情報を上書きする(S1004)。

【0093】このような処理を、ステップS1004において行なう。

【0094】次に、カラム情報記憶部9へ登録を終えた後、カラム出力受付部8は、カラム要求部10に対し、カラム転送を指示する。カラム要求部10は、カラム番号とデータ型をパラメータとして有するカラム出力要求を、データベース管理部3へ発行し(S1005)、転送された1つのカラムデータを、アプリケーション1によって指定されたバッファアドレス27が指す領域に格納した後、カラムデータの長さを、指定されたデータ長アドレス28が指す領域に格納する(S1006)。な

お、バッファアドレス27が指す領域やデータ長アドレス28が指す領域は、アプリケーション1内の特定のエリアに設けられるようにしておけばよい。

【0095】一方、S1003において、要求内容から抽出したカラム番号25と同一の値が登録されたカラム番号記憶領域18が存在し、かつ、そのカラム番号記憶領域18に対応するデータ型記憶領域19の値が、要求内容から抽出したデータ型26と同一である場合、カラム出力受付部8は、まず、ブロックバッファ12aを参照して(S1007)、抽出したカラム番号25と同一の値が登録されたカラム番号20を探す。次に、探し出したカラム番号20に対応するカラムデータ22を、アプリケーション1によって指定されたバッファアドレス27が指す領域に格納し、さらに、カラムデータの長さ21を、指定されたデータ長アドレス28が指す領域に格納する(S1008)。

【0096】図11は、データベース管理部3におけるカラム転送部15の動作を示すフローチャートである。

【0097】カラム転送部15は、アクセス制御部2から、カラム出力要求を受け付け(S1101)、要求内容を表示情報であるパラメータである、カラム番号25とデータ型26とを抽出し、夫々、カラム情報記憶部9bのカラム番号記憶領域18、データ型記憶領域19に登録する。

【0098】抽出したカラム番号25と同一の値が登録されたカラム番号記憶領域18が存在する場合には、抽出したデータ型26を、データ型記憶領域19に上書きする(S1102)。

【0099】次に、カラム転送部15は、カラム変換出力部16に対して、要求内容から抽出したカラム番号25とデータ型26を通じて、カラムデータを変換して出力する旨の指示を行なう。カラム変換出力部16は、指定されたカラムのデータを、データベース4から取り出し、指示されたデータ型に変換して、カラム転送部15へ返す処理を行なう(S1103)。カラム転送部15は、カラム変換出力部16から受け取ったカラムデータを、アクセス制御部2へ転送する(S1104)。

【0100】以上が、カラム転送部15が行なう動作内容である。

【0101】図12は、データベース管理部3におけるブロック転送部17が行なう動作内容を示すフローチャートである。

【0102】まず、ブロック転送部17は、アクセス制御部2から、ブロック出力要求を受け付け(S1201)、カーソル制御部14に對して、カーソル移動要求を発行する(S1202)。次に、カーソル制御部14は、ブロック転送部17、または、アクセス制御部2のカーソル移動受付部7から、カーソル移動の要求を受けて、導出表中の現在のカーソルを、次のレコードに進める。

【0103】次に、ブロック転送部17は、カラム情報記憶部9bに登録された最初のカラムに関する属性を参照し(§1203)、カラム変換出力部16に対して、カラム番号記憶域18とデータ型記憶域19に登録された値を通知して、カラムデータが変換して出力されるように指示する。

【0104】カラム変換出力部16は、指示されたカラムデータをデータベース4から取り出し、指示されたデータ型に変換して、ブロックバッファ12bの所定領域に格納する(§1204)。さらに、ブロック転送部17は、カラム情報記憶部9bを参照して、全てのカラムデータを出力したか否かを確認する(§1205)。出力していないカラムデータが存在する場合、カラム情報記憶部9bに登録された、次カラムデータに関する属性を参照し(§1205)、カラム変換出力部16に対して、次カラムデータの変換出力を指示する(§1204)。

【0105】ブロック転送部17は、ステップ§1204のカラム変換出力処理を、カラム情報記憶部9bに登録された、全てのカラムデータに対して繰り返した後、ブロックバッファ12bに格納された、全てのカラムデータをアクセス制御部2へ転送する処理を行なう(§1207)。

【0106】以上、本発明にかかるデータベースアクセスシステムの一実施例について説明した。

【0107】従来のデータベースアクセスシステムにおけるデータベース管理部3は、アプリケーション1から与えられるSQLGetCol関数3の属性とは無関係に、導出曲中の複数のカラムデータを一括して転送するため、データの折落ち、丸め等の発生、冗長なデータ転送処理によるオーバヘッドの増加等の問題点が存在していた。

【0108】これに対し、本発明にかかるデータベースアクセスシステムでは、データベース管理部3が、アプリケーション1から与えられたSQLGetCol関数3の属性を記憶しておき、アプリケーション1から出力要求されたカラムに対するデータだけを、指定されたデータ型に変換処理して転送するため、データの折落ち、丸め、冗長なデータ転送動作等の無い、高速かつ高信頼性を有する、データベースシステムを構築することが可能となる。

【0109】なお、本発明は、アクセス制御部2が動作するクライアントと、データベース管理システム3が動作するサーバからなる、クライアントサーバシステムとして提供可能である。

【0110】一方、マルチベンダの異なるメーカーにより製造されたデータベース管理部3に対して、同一のアプリケーション1から、アクセス可能とするために、ゲートウェイ装置を導入する場合も多い。

【0111】本発明は、このようなシステム構成にも適用可能である。

【0112】図14に、本発明にかかるデータベースアクセスシステムの、他の実施例のシステム構成を示す。

【0113】なお、図1と比べて分かるように、主たる構成要素は、図1に示すシステムと変わることろがないが、本実施例では、新たにゲートウェイ23を設け、言語変換部24を設けるとともに、図1のシステムでは、データベース管理部3内に設けられていた、通信制御部5b、ブロック転送部17、カラム転送部15、ブロックバッファ12bおよびカラム情報記憶部9bを、ゲートウェイ側に配置している。

【0114】なお、図1に示した構成要素と同一の構成要素の機能や構成に変わることろはない。

【0115】ゲートウェイ23は、マルチベンダのデータベース管理部間でのSQLの差異を吸収する機能を有する処理手段であり、本実施例では、データベース管理部3およびデータベース4と同一の情報処理装置上で動作する。もちろん、ゲートウェイ23を、アクセス制御部2、データベース管理部3、およびデータベース4と、異なる情報処理装置で構成して動作させることも可能である。

【0116】図14において、アクセス制御部2は、前述した構成と変わることろがない。

【0117】前述したように、ゲートウェイ23は、通信制御部5b、ブロック転送部17、カラム転送部15、ブロックバッファ12b、カラム情報記憶部9b、および、言語変換部24を有して構成されている。言語変換部24以外は、図1に示したものと変わることろがない。

【0118】言語変換部24は、アクセス制御部2から受けた標準SQLを、データベース管理部固有のSQLに変換する機能を有している。すなわち、標準SQLを、アクセス対象となる、ベンダーごとのデータベース管理部に対する固有の処理言語に変換する処理を行なっている。これにより、ユーザは、アプリケーションによるアクセス対象となるデータベース管理部固有の処理言語を意識せずに、標準SQLを使用することができるため、利便性に富む。

【0119】なお、前述したように、各構成要素が行なう動作は、図9から12で示したものと同一であるため、詳しく述べないが、概要のみを説明しておくと、次のようになる。

【0120】ゲートウェイ23のカラム転送部15は、アクセス制御部2から受けたカラム出力要求を解析し、カラム番号と出力時のデータ型を、カラム情報記憶部9bに記憶する。ゲートウェイ23のブロック転送部17は、アクセス制御部2から、ブロック出力要求を受け、カラム情報記憶部9bに記憶したカラム番号とデータ型を参照して、データベース4上のカラムデータを変換して、ブロックバッファ12bに格納し、接続のカラムデータを連結して、アクセス制御部2へ返す。なお、

データベース操作部13の動作は、言語変換部24によって変換された言語によって制御されることになる。

【0122】以上のように、本実施例によれば、アクセス制御部2とデータベース管理部3の間に、ゲートウェイ23を配置した構成においても、データの桁落ち、丸め、冗長なデータ転送処理等の無い、高速かつ高信頼を有するデータベースアクセスシステムを構築できることになる。

【0122】なお、以上説明したきた実施例では、1レコード中の全カラムデータを一括して転送する方法について説明したが、複数レコードの全カラムデータを一括転送するより動作を制御してもよい。また、1レコードの全カラムデータが、ロックバッファ12の記憶領域を越える長さである場合には、幾つかのロックに分割して、カラムデータを転送するように制御する構成としても良い。

【0123】さらに、必要な構成を、クライアント側とサーバー側に分けて配置して、クライアント・サーバシステムを提供するにしてもよいことは言うまでもない。

【0124】

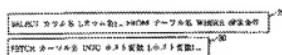
【発明の効果】以上説明してきたように、本発明によれば、高速でデータベースをアクセス可能なシステムを実現できる。また、アプリケーション等から要求されたカラムデータだけを、要求されたデータ形式で出力するため、データの桁落ち、丸め誤差、冗長なデータ転送等が無い、高性能なデータベースアクセスシステムを構築できる。

【図1】本発明の第1実施例の構成図である。

【図2】SQL構文のシnta克斯の一例を示す説明図である。

【図2】

図 2



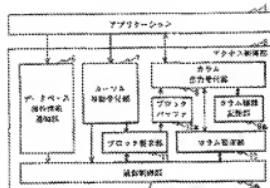
【図6】

図 6



【図1】

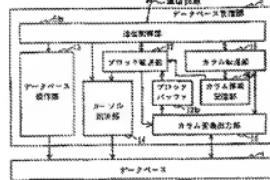
図 1



【図4】

図 4

アブリューション	アクセス制御部	データベース実行部
SQLExec	アクセス権限	データソース登録
SQLFetch	データソース登録	カラム圧力選択
SQLGetCol	カラム圧力選択	カラム圧力選択
SQLGetText	カラム圧力選択	カラム圧力選択
SQLInsert	カラム圧力選択	カラム圧力選択
SQLUpdate	カラム圧力選択	カラム圧力選択
SQLDelete	カラム圧力選択	カラム圧力選択
SQLDrop	カラム圧力選択	カラム圧力選択
SQLAlter	カラム圧力選択	カラム圧力選択
SQLGetCell	カラム圧力選択	カラム圧力選択
SQLSetCell	カラム圧力選択	カラム圧力選択
SQLGetVal	カラム圧力選択	カラム圧力選択
SQLSetVal	カラム圧力選択	カラム圧力選択
...



【図5】

図 5

アブリューション	アクセス制御部	データベース実行部
SELECT	アクセス権限	ロック圧力選択
FETCH	ロック圧力選択	カラム圧力選択
カラム選択	カラム圧力選択	カラム圧力選択
カラム削除	カラム圧力選択	カラム圧力選択
DELETE	ブロック化要求	ブロック化要求
カラム削除	ブロック化要求	カラム削除
カラム更新	ブロック化要求	カラム削除
カラム追加	ブロック化要求	カラム追加
カラム変更	ブロック化要求	カラム変更
カラム削除	ブロック化要求	カラム削除
...

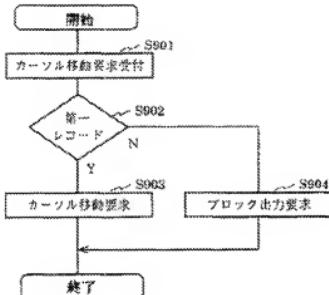
【図8】

図 8

アブリューション	アクセス制御部	データベース実行部
SQLExec	アクセス権限	データソース登録
SQLFetch	データソース登録	カラム圧力選択
SQLGetCol	カラム圧力選択	カラム圧力選択
SQLGetText	カラム圧力選択	カラム圧力選択
SQLInsert	カラム圧力選択	ブロック化要求
SQLUpdate	カラム圧力選択	ブロック化要求
SQLDelete	カラム圧力選択	ブロック化要求
SQLDrop	カラム圧力選択	ブロック化要求
SQLAlter	カラム圧力選択	カラム圧力選択
SQLGetCell	カラム圧力選択	カラム圧力選択
SQLSetCell	カラム圧力選択	カラム圧力選択
SQLGetVal	カラム圧力選択	カラム圧力選択
SQLSetVal	カラム圧力選択	カラム圧力選択
...

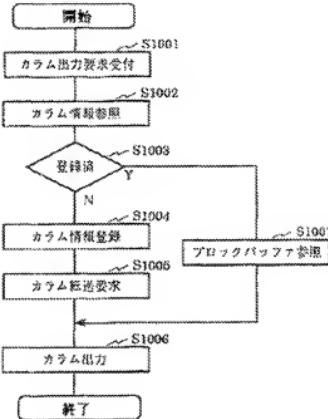
《國學》

90



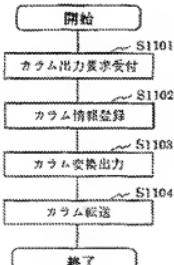
[Page 10]

圖 10



卷之三

三



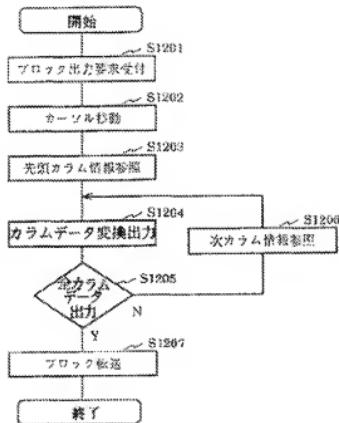
[图 1-3]

图 13



【図12】

図 1 2



【図14】

図 1 4

